

Sub  
AI

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25

**CLAIMS**

1. A method comprising:  
loading a reflection image into memory;  
retrieving an environment texture sample from an environment map based  
on a reflection vector stored in a pixel of the reflection image; and  
applying the environment texture sample to the object.
2. The method of claim 1, wherein the loading comprises:  
retrieving a reflection texture sample comprising red, green, and blue color  
data; and  
storing the red, green, and blue color data of the reflection texture sample as  
red, green, and blue color data of a pixel of the object.
3. The method of claim 2, wherein the retrieving comprises interpreting  
the red, green, and blue color data of the pixel as the reflection vector.
4. The method of claim 3, wherein the retrieving comprises retrieving  
the environment texture sample comprising red, green, and blue color data from  
the environment map based on the interpreted reflection vector.
5. The method of claim 4, wherein the applying comprises replacing  
the red, green, and blue color data of the pixel with the red, green, and blue color  
data of the environment texture sample.

1        6.     The method of claim 3, further comprising perturbing the interpreted  
2 reflection vector prior to retrieving the environment texture sample.

3  
4        7.     The method of claim 1, wherein the loading, the retrieving, and the  
5 applying are performed during a single pass through a graphics pipeline.

6  
7        8.     The method of claim 6, further comprising storing a result in a frame  
8 buffer.

9  
10       9.     The method of claim 1, wherein the loading is performed during a  
11 first pass through a graphics pipeline and the retrieving and the applying are  
12 performed during a second pass through the graphics pipeline.

13  
14       10.    The method of claim 9, further comprising:  
15 storing the reflection image in a frame buffer; and  
16 replacing the reflection image in the frame buffer with a result of  
17 application of the environment texture sample.

18  
19       11.    The method of claim 10, further comprising:  
20 loading the reflection image in a texture memory; and  
21 loading the environment map in the texture memory prior to performing the  
22 retrieving and the applying.

23  
24       12.    The method of claim 1, wherein the retrieving comprises retrieving  
25 the environment texture sample from a cube environment map.

1  
2 13. The method of claim 1, further comprising:  
3 generating a plurality of reflection images, wherein each of the plurality of  
4 reflection images corresponds to a particular viewpoint; and  
5 loading a predetermined reflection image chosen from the plurality of  
6 reflection images into the memory.

7  
8 14. The method of claim 1, wherein the loading, the retrieving, and the  
9 applying are performed in real time.

10  
11 15. A method comprising  
12 retrieving a texture sample from a texture map, the texture sample  
13 containing reflection data;  
14 using the reflection data in the texture sample to obtain an environment  
15 texture sample in an environment map; and  
16 applying the environment texture sample to an object.

17  
18 16. The method of claim 15, wherein the applying comprises replacing  
19 red, green, and blue color data of the pixel with red, green, and blue color data of  
20 the environment texture sample.

21  
22 17. The method of claim 15, further comprising perturbing the reflection  
23 data prior to using the reflection data to obtain the environment texture sample.

1        18. The method of claim 15, wherein the using comprises using the  
2 reflection data to obtain an environment texture sample in a cube environment  
3 map.

4  
5        19. The method of claim 15, further comprising storing, in memory, a  
6 result of said applying the environment texture sample to the object.

7  
8        20. A method comprising:  
9 interpreting data in a pixel as a reflection vector; and  
10 using the reflection vector to obtain an environment texture sample in an  
11 environment map.

12  
13        21. The method of claim 20, wherein the interpreting comprises  
14 interpreting red, green, and blue color data of the pixel as the reflection vector.

15  
16        22. The method of claim 20, further comprising perturbing the reflection  
17 vector prior to using the reflection vector to obtain the environment texture  
18 sample.

19  
20        23. The method of claim 20, wherein the using comprises using the  
21 reflection data to obtain an environment texture sample in a cube environment  
22 map.

23  
24        24. The method of claim 20, further comprising applying the  
25 environment texture sample to an object.

1  
2 25. A method comprising:  
3 retrieving a texture sample from a texture map; and  
4 using the texture sample to retrieve an environment texture sample from an  
5 environment map.

6  
7 26. A computer program product comprising a computer useable  
8 medium having computer program logic recorded thereon for enabling a processor  
9 to render a computer scene, the computer program logic comprising:

10 a texture map comprising reflection data;

11 a texture map sampling procedure that enables the processor to obtain a  
12 first texture sample from the texture map and apply the first texture sample to an  
13 object;

14 an environment map; and

15 an environment map sampling procedure that enables the processor to  
16 obtain a second texture sample from the environment map based on the first  
17 texture sample and apply the second texture sample to the object.

18  
19 27. The computer program product of claim 26, wherein the texture map  
20 sampling procedure enables the processor to obtain red, green, and blue color data  
21 from the texture map and store the red, green, and blue color data as a pixel of the  
22 object.

23  
24 28. The computer program product of claim 27, wherein the  
25 environment map sampling procedure enables the processor to use the red, green,

1 and blue color data of the pixel as a reflection vector to obtain the second texture  
2 sample.

3  
4 29. The computer program product of claim 26, wherein the  
5 environment map comprises a cube environment map.

6  
7 30. The computer program product of claim 26, further comprising a  
8 texture map generating procedure that enables the processor to generate a  
9 particular texture map comprising reflection data based on a particular viewpoint.

10  
11 31. A computer readable medium storing computer-readable instructions  
12 that, when executed, direct a graphics processing system to:

13 interpret data in one of a reflection image or a texture map as a reflection  
14 vector;

15 use the reflection vector to obtain an environment texture sample in an  
16 environment map; and

17 apply the environment texture sample to an object.

18  
19 32. The computer readable medium of claim 31, further comprising  
20 computer-readable instructions that, when executed, direct the graphics processing  
21 system to interpret red, green, and blue color data as the reflection vector.

22  
23 33. The computer readable medium of claim 31, further comprising  
24 computer-readable instructions that, when executed, direct the graphics processing  
25

1 system to perturb the reflection vector prior to using the reflection vector to obtain  
2 the environment texture sample.

3  
4 34. A system for environment mapping, comprising:  
5 an application program having computer program logic, an environment  
6 map, and a texture map comprising reflection data;  
7 a processor for implementing the computer program logic; and  
8 a graphics subsystem, under the control of the application program and the  
9 processor, for rendering an object using the texture map and the environmental  
10 map.

11  
12 35. The system of claim 34, wherein the graphics subsystem comprises:  
13 a texture memory;  
14 a frame buffer; and  
15 a rasterizer coupled to the texture memory and the frame buffer.

16  
17 36. The system of claim 34, wherein the graphics subsystem is  
18 configured to use a texture sample retrieved from the texture map to obtain an  
19 environment texture sample from the environment map and to apply the  
20 environment texture sample to the object.

21  
22 37. The system of claim 34, wherein the graphics subsystem is  
23 configured to retrieve an environment texture sample from the environment map  
24 based on a reflection vector stored in a texture sample from the texture map and to  
25 apply the environment texture sample to the object.

Concl  
AI

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25

38. The system of claim 34, wherein the graphics subsystem is configured to retrieve a texture sample from the texture map, the texture sample containing reflection data, the graphics subsystem being further configured to use the reflection data in the texture sample to obtain an environment texture sample in the environment map.